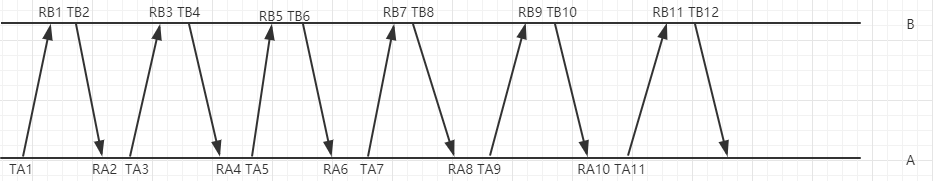
1. 前提：
   1. 每台无人机一直在接收其他无人机的报文，接收后无人机将更新本地信息。并进行测距计算
   2. 每过一段时间(周期)，无人机会发一个报文。
   3. 每台无人机需要存储最近N个周期，发出报文的时间戳。
   4. 每台无人机需要保存其他无人机最近一次的历史报文。
2. 测距流程
   1. One2One

假设目前有2台无人机为A,B，T表示为传送，R为接收，并用标号每次传输则自增1。则有以下流程：



每台无人机的 T 与 R 的功能相同。

对A来说，假设现在进行第 N 次传送(TAN), 则该次传送包含的内容为一个四元组:

(Idx\_A, Idx\_B, t ( TA(N-2) ), t( RA(N-1) ) )

Idx\_TA: 本次传送的标识索引,Idx\_A = N

Idx\_TB: 上一次B传输过来的标识索引

t(X): 获取时间戳

t( TA(N-2) ): 上次传送产生的时间戳

t( RA(N-1) ): 本次接收信息产生的时间戳

在第 N 次 数据传送完成后，A将进行测距，测距的序列为：

TA(N-4) -> RB(N-4)->TB(N-3)->RA(N-3)->TA(N-2)->RB(N-2)

**以上序列在第N次测距时都已知。带入测距公式即可得到距离。**

例子：

现在， 将 N = 9 带入该序列 则可以获得：

TA5-> RB5->TB6->RA6->TA7->RB7

说明：

RA6, TA5, TA7: 保存在本地的时间戳

TB6，RB7：由 TB8 带来的三元组获得

RB5 : 由TB6带来的三元组获得

在计算完距离后，更新本地信息。

更新信息:

为了方便理解，用大小为2\*5的数组抽象表示历史信息与本次获得的信息。

修改前信息,即计算距离前,无人机A上能找到的所有关于和B的测距信息:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | Idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(unknown) | 6(known) | RA6(known) | 5(known) |
| 2 | (unknown) | (unknown) | (unknown) | (unknown) | (unknown) |

收到了来自B给与的信息,并打上A本地的时间戳RA8：

获得了B带来的信息, idx\_TB = 8,TB6,RB7, Idx\_TA = 7

修改后信息:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(known) | 6(known) | RA6(known) | 5(known) |
| 2 | RB7(known) | TB8(unknown) | 8(known) | RA8(known) | 7(known) |

进行计算。

计算完以后，用第二行把第一行覆盖掉。

重复以上过程。

* 1. One2Many

一对多的测距是由一对一测距的扩展。

对于无人机 A，本地需要针对其他无人机进行类似的缓存过程。因此，可以抽象成N\*2\*5的数组。

每台无人机发送的消息则变成了三元组+本次的索引以及上一次的时间。

对A来说，假设现在进行第 N 次传送(TAN)

|  |  |
| --- | --- |
| 本次索引 | Idx\_TA |
| 上次发送时间 | t ( TA(Idx\_TA -2) ) |
| (本次收到的无人机代号,接收的索引, 本次接收时间戳) | ( B, Idx\_TB, tb),  (C,Idx\_TC , tc),  … |

1. 错误处理(丢包，重复接收)

针对一对一情况进行说明，一对多情况类似。简化说明就以 A 与 B 测距为例子。

1. 丢包

情况一：(假丢包)

设在N= 7时，A发送了报文。

设在 N= 9 时，A没有收到B的报文。B尚在处理

设在N=11时，A收到了B的报文，B应答的是N=7(Idx\_TA=7)的情况。

情况二：(真丢包)

设在N= 7时，A发送了报文。A报文丢失。

设在N= 9 时，A没有收到B的报文。

设在N=11时，A收到了B的报文，B应答的是N=9(Idx\_TA=9)的情况。

情况三：(真丢包)

设在N= 7时，A发送了报文。A报文未丢失。

设在N= 9 时，A没有收到B的报文。B报文丢失，B已经过了一个周期

设在N=11时，A收到了B的报文，B应答的是N=9(Idx\_TA=9)的情况。

情况四：  
设在N= 7时，A发送了报文。

设在 N= 9 时，A没有收到B的报文。B尚在处理

设在N=11时，A收到了B的2个报文

尚未想好如何处理。。。2020/1/4

策略：

只对发送期间收到报文的无人机进行更新。

对于情况一:

在N=9时，则不做任何更新操作。

在N=11时,对B的信息依然停留在上次。

更新前：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | Idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(unknown) | 6(known) | RA6(known) | 5(known) |
| 2 | (unknown) | (unknown) | (unknown) | (unknown) | (unknown) |

收到了来自B给与的信息,并打上A本地的时间戳RA8：

获得了B带来的信息, idx\_TB = 8,TB6,RB9, Idx\_TA=7

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(known) | 6(known) | RA6(known) | 5(known) |
| 2 | RB7(known) | TB8(unknown) | 8(known) | RA8(known) | 7 (known) |

当N=11时, 本来应当测距序列为:

TA(7) -> RB(7)->TB(8)->RA(8)->TA(9)->RB(9)

现在改为:

TA(5) -> RB(5)->TB(6)->RA(6)->TA(7)->RB(7) (即A在N=11 时，做了N =9 时的操作)

结论：假丢包会导致测距的延时加长，延时时间与B的处理时间有关，对相对静止的无影响。不会造成对整体过程的不可逆损坏，可以保证测距继续进行。

对于情况二：

在N=9时，则不做任何更新操作。

在N=11时,

更新前：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | Idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(unknown) | 6(known) | RA6(known) | 5(known) |
| 2 | (unknown) | (unknown) | (unknown) | (unknown) | (unknown) |

收到了来自B给与的信息,并打上A本地的时间戳RA8：

获得了B带来的信息, idx\_TB = 8,TB6,RB7, Idx\_TA=9

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(known) | 6(known) | RA6(known) | 5(known) |
| 2 | RB9(known) | TB8(unknown) | 8(known) | RA8(known) | 9(known) |

当N=11时, 本来应当测距序列为:

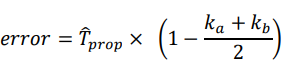
TA(7) -> RB(7)->TB(8)->RA(8)->TA(9)->RB(9)

现在改为:

TA(5) -> RB(5)->TB(6)->RA(6)->TA(9)->RB(9)

结论：

根据误差公式可知测距的误差只与传播时间与晶振频率相关。



异步测距不会造成误差。只要在该段测距时间内(TA(5)----->RB(9) )的物体保持相对静止则可以保证误差并非算法造成。不会造成对整体过程的不可逆损坏，即可以保证测距继续进行。

对于情况三:

在N=9时，则不做任何更新操作。

在N=11时,对B的信息依然停留在上次。

更新前：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | Idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(unknown) | 6(known) | RA6(known) | 5(known) |
| 2 | (unknown) | (unknown) | (unknown) | (unknown) | (unknown) |

收到了来自B给与的信息,并打上A本地的时间戳RA8：

获得了B带来的信息, idx\_TB = 10**,TB8,**RB10, Idx\_TA=9

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | TB | idx\_TB | RA | Idx\_TA |
| 1 | RB5(known) | TB6(**unknown**) | 6(known) | RA6(known) | 5(known) |
| 2 | RB9(known) | TB10(unknown) | 10(known) | RA8(known) | 9(known) |

**此时发现无法计算，TB6未知，不做计算，只更新本地数值(即用第二行把第一行覆盖掉)。**

结论：舍弃此次更新。该次丢包不会影响到下一次。由于丢包导致的是测距的延时加长，不确定性增加。不会造成对整体过程的不可逆损坏，可以保证测距继续进行。

1. 重复接收

考虑这样一个情况。 A的传输周期远远大于B。导致A在一个周期内部收到了来自B的2次传送的报文。

显然A 收到的包是 由B顺序发出的。

例子：

设在N= 7时，A发送了报文。

设在 N= 9 时，A没有收到B的报文，A继续发送报文。

场景一：

设在N=11时，A收到了B的2个报文，B两次应答分别是N=7(Idx\_TA=7)以及N=9(Idx\_TA=7)。

可以将情况归类为 丢包场景一(假丢包)。

场景二：

设在N=11时，A收到了B的2个报文，B两次应答分别是N=7(Idx\_TA=7)以及B的报文未涉及A。

可以将情况归类为 丢包场景一(假丢包)，以及不做处理。

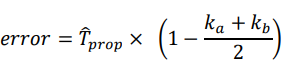
场景三：

设在N=11时，A收到了B的2个报文，B的两次应答分别是B的报文未涉及A 和N=9(Idx\_TA=7)。

可以将情况归类为不做处理，丢包场景一(假丢包)

1. 误差引入

如前文提到，TWR误差主要的来源是传播时间与晶振频率。



涉及过程中引入的误差主要来源于异步性造成的误差。

* + - 1. 测距过程为非实时测距(相差一个周期)
      2. 丢包以及重复接收造成非实时测距可能会有多个周期的延时。

1. 测距方式
   1. 直接测距

直接套用测距公式：TWR (t)

误差分析：

误差主要由误差公式造成，没有进行多次测距，测距值波动较大。在测试相对静止物体时无法提高精度。对速度较快的物体测距，实时性较好，计算量小，不受历史结果影响。

1. 缺点
   1. 会不会有信道占用，导致丢包或错误的问题？使用随机数进行传输以减小丢包概率
   2. 如果使用迭代测距的方式，不知道何时收敛。不使用迭代
   3. 接近实时，依然存在一个时间差。